Generating Visual and Interactive Output from System Engineering Tools

John Schatz / SPEC Innovations
29 October 2009
Overview

- Methodology
- KBAD Schema
- Information Capture Process
- Risk Matrix Visualization
- TPM Capture and Visualization
- Expandable-Collapsible Tree Visualization
- Geo-Spatial Visualization
Methodology

Modify SE Knowledge-Base Schema

Capture Information in SE Knowledge-Base

Extract Data from SE Knowledge-base

Execute Output Loop

Execute Criteria Based Business Logic

Output Loop

Export Data in Modified Format (RTF, HTML, XML)

Determine Outputs
The KBAD Schema supports the capture of data items and relationships utilized in the examples.

*Knowledge-Based Analysis and Design*
Capture Information in SE Knowledge-Base

1. Capture and Analyze Related Documents
2. Identify Assumptions
3. Identify Existing/Planned Systems
4. Capture Constraints
5. Develop the Operational Context Diagram
6. Develop Operational Scenarios
7. Derive Functional Behavior
8. Derive System Elements
9. Allocate Functions to System Elements
10. Prepare Interface Diagrams
11. Define Resources, Error Detection & Recovery
12. Perform Dynamic Analysis
13. Develop Operational Demonstration Master Plan
14. Provide Options
15. Conduct Trade-off Analyses
16. Generate Operational and System Architecture Graphics, Briefings and Reports

Time

The KBAD middle-out approach has been proven on a variety of projects.
Risk Matrix Example

Figure 1  Risk Characterization Summary Matrix

Consequence Impact: C1 = Negligible, C2 = Minor, C3 = Moderate, C4 = Serious, C5 = Critical
Likelihood: L1 = (0-10), L2 = (10-40), L3 = (40-60), L4 = (60-90), L5 = (90-100)
Risk Characterization: Green = Low, Yellow = Moderate, Red = High
Risk Matrix Example - Logic

1. Extract Risks of interest.
2. Create lists of risks for each Risk Matrix cell by examining the risks’ consequences and likelihoods.
3. Begin streaming RTF file up to first cell.
4. Set cell color. The cell colors are fixed.
5. Insert risks for given cell.
### TPM Example

#### Technical Performance Measures

<table>
<thead>
<tr>
<th>Hardware Assets</th>
<th>Current</th>
<th>Projected</th>
<th>Threshold</th>
<th>Objective</th>
<th>Units</th>
<th>Imp.Dir.</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>SBPG Ground Element</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>TPM: Ground Element MTTR</td>
<td>60.0</td>
<td>31.0</td>
<td>30.0</td>
<td>15.0</td>
<td>minutes</td>
<td>Negative</td>
</tr>
<tr>
<td><strong>SBPG On-Orbit Element</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>TPM: On-Orbit Element Lifespan</td>
<td>13.0</td>
<td>15.2</td>
<td>10.0</td>
<td>15.0</td>
<td>years</td>
<td>Positive</td>
</tr>
<tr>
<td>TPM: On-Orbit Element Transmission Efficiency</td>
<td>0.63</td>
<td>0.65</td>
<td>0.65</td>
<td>0.75</td>
<td>kg</td>
<td>Positive</td>
</tr>
<tr>
<td>TPM: On-Orbit Element Weight</td>
<td>6500.0</td>
<td>5800.0</td>
<td>6000.0</td>
<td>5000.0</td>
<td>kg</td>
<td>Negative</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>System Functions</th>
<th>Execute Maneuver Commands</th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>TPM: Characteristic_001</td>
<td>10.0</td>
<td>13.2</td>
<td>11.0</td>
<td>13.0</td>
<td>seconds</td>
<td>Positive</td>
</tr>
<tr>
<td>Issue Maintenance Alert</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>TPM: Characteristic_002</td>
<td></td>
<td></td>
<td>0.95</td>
<td>0.98</td>
<td>Positive</td>
<td></td>
</tr>
<tr>
<td>Collect Solar Energy</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>TPM: Characteristic_003</td>
<td>0.65</td>
<td>0.66</td>
<td>0.6</td>
<td>0.55</td>
<td>Positive</td>
<td></td>
</tr>
</tbody>
</table>

**Figure 1** SBPG Context Performance Parameters Matrix

Value Characterization: Green = Exceeds Objective, Yellow = Between Threshold and Objective, Red = Does Not Meet Threshold
TPM Example - Logic

1. Extract TPMs for systems of interest.
2. Begin streaming RTF file up to first System row.
3. Insert System name.
4. Stream up to the system’s first TPM.
5. Insert TPM name.
6. Compare current and projected values against threshold and object values taking improvement direction into account.
7. Determine cell color based on predetermined criteria.
8. Insert color coded cells with current and projected values.
Expandable-Collapsible Tree Example
Expandable-Collapsible Tree Example - Logic

1. Write JavaScript and CSS files.
2. Extract nodes in an interconnected nodal network.
3. Generate index file of all nodes.
4. Iterate through nodes doing the following for each:
   1. Generate expandable tree branches and leaves. Prevent closed loops.
   2. Begin streaming HTML file with JavaScript and CSS files referenced.
   3. Store controls and tree data in JavaScript node array.
   4. Store starting positions in JavaScript position array.
   5. Encode nodes into HTML file as absolutely positioned items with embedded JavaScript commands to access Document Object Model (DOM) for hiding or showing nodes.
Geo-Spatial Example
Geo-Spatial Example - Logic

1. Generate KML header information.
2. Extract assets from SE Knowledge-Base.
3. Iterate through assets streaming asset specific KML.
Summary

• Use of other products for visualization is necessary, since most SE tools provide poor graphics for a general audience
• Output from COTS Products can be modified to enhance visualization
• Most tools provide scripting that enable creative visualization